Solutions - Midterm Exam

(Due date: February 15th)

Presentation and clarity are very important! Show your procedure!

PROBLEM 1 (20 PTS)

- Compute the result of the following operations. The operands are signed fixed-point numbers. The result must be a signed fixed point number. For the division, use $x = 5$ fractional bits.

1.0001 + 1.001001	1000.0101 − 1.010101	01.11111 + 0.00001
01.011 × 1.01101	1.001 × 1.0101	01.01110 ÷ 1.011

$c_8=1$ $c_7=1$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

1 1.0 0 0 1 0 0 +
1 1.0 0 1 0 0 1
1 0.0 0 1 1 0 1

1 0 0 0.0 1 0 1 0 0 −
1 1 1 1.0 1 0 1 0 1
➡
$c_{10}=0$ $c_9=0$ $c_8=0$ $c_7=0$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

1 0 0 0.0 1 0 1 0 0 +
0 0 0 0.1 0 1 0 1 1
1 0 0 0.1 1 1 1 1 1

$c_8=0$ $c_7=0$ $c_6=1$ $c_5=1$ $c_4=1$ $c_3=1$ $c_2=1$ $c_1=1$ $c_0=0$

0 0 1.1 1 1 1 1 1 +
0 0 0.0 0 0 0 0 1
0 1 0.0 0 0 0 0 0

01.011 × 1.01101 ➡ 01.011 × 0.10011 ➡

1 0 0 1 1 ×
1 0 1 1
1 0 0 1 1
1 0 0 1 1
0 0 0 0 0
1 0 0 1 1
0 1 1 0 1 0 0 0 1
⬇
0.1 1 0 1 0 0 0 1
⬇
1.0 0 1 0 1 1 1 1

1.001 × 1.0101 ➡ 0.111 × 0.1011 ➡

1 0 1 1 ×
1 1 1
1 0 1 1
1 0 1 1
1 0 1 1
0 1 0 0 1 1 0 1
⬇
0.1 0 0 1 1 0 1

✓ $\frac{01.01110}{1.011}$: To unsigned (denominator) and then alignment, $a = 4$: $\frac{01.0111}{0.101} = \frac{01.0111}{00.1010} = \frac{010111}{001010} \equiv \frac{10111}{1010}$

0001001001
1010) 1011100000
1010
1100
1010
10000
1010
110

Append $x = 5$ zeros: $\frac{10111\mathbf{00000}}{1010}$

Integer Division:

$Q = 1001001, R = 110$
$\rightarrow Qf = 10.01001 \ (x = 5)$

Final result (2C): $\frac{01.01110}{1.011} = 2C(010.01001) = 101.10111$

PROBLEM 2 (12 PTS)

- Represent these numbers in Fixed Point Arithmetic (signed numbers). Select the minimum number of bits in each case.
  - ✓ -16.3125
    +16.3125 = 010000.0101
    ⇒ -16.3125 = 101111.1011
  - ✓ 37.375
    +37.375 = 0100101.011

- Complete the table for the following fixed point formats (signed numbers): (6 pts.)

Integer bits	Fractional Bits	FX Format	Range	Resolution
6	2	[8 2]	$[-2^5, 2^5 - 2^{-2}]$	2^{-2}
8	4	[12 4]	$[-2^7, 2^7 - 2^{-4}]$	2^{-4}

PROBLEM 3 (36 PTS)

- Calculate the result (provide the 32-bit result) of the following operations with 32-bit floating point numbers. Truncate the results when required. When doing fixed-point division, use 4 fractional bits. Show your procedure.

✓ 42FA8000 + C0E00000	✓ 50DAD000 - D0FAD000	✓ 01800000 × FAB80000	✓ 7B390000 ÷ C8C00000

✓ X = 42FA8000 + C0E00000:

42FA8000: 0100 0010 1111 1010 1000 0000 0000 0000

$e + bias = 10000101 = 133 \rightarrow e = 133 - 127 = 6$ $\quad$ $Significand = 1.11110101$

42FA8000 = 1.11110101×2^6

C0E00000: 1100 0000 1110 0000 0000 0000 0000 0000

$e + bias = 10000001 = 129 \rightarrow e = 129 - 127 = 2$ $\quad$ $Significand = 1.11$

C0E00000 = -1.11×2^2

$X = 1.11110101 \times 2^6 - 1.11 \times 2^2 = 1.11110101 \times 2^6 - \frac{1.11}{2^4} \times 2^6$

$= (1.11110101 - 0.000111) \times 2^6$

```
c10=1 c9=1 c8=1 c7=1 c6=1 c5=0 c4=0 c3=1 c2=0 c1=0 c0=0
  0 1.1 1 1 1 0 1 0 1 +
  1 1.1 1 1 0 0 1 0 0
  -------------------
  0 1.1 1 0 1 1 0 0 1
```

To subtract these unsigned numbers, we first convert to 2C:

$R = 01.11110101 - 0.000111 = 01.11110101 + 1.111001$

The result in 2C is: $R = 01.11011001$

For floating point, we need to convert to sign-and-magnitude:

$\Rightarrow R(SM) = +1.11011001$

$X = 1.11011001 \times 2^6$, $e + bias = 6 + 127 = 133 = 10000101$

X = 0100 0010 1110 1100 1000 0000 0000 0000 = 42EC8000

✓ X = 50DAD000 - D0FAD000:

50DAD000: 0101 0000 1101 1010 1101 0000 0000 0000

$e + bias = 10100001 = 161 \rightarrow e = 161 - 127 = 34$ $\quad$ $Significand = 1.10110101101$

10DAD000 = $1.10110101101 \times 2^{34}$

D0FAD000: 1101 0000 1111 1010 1101 0000 0000 0000

$e + bias = 10100001 = 161 \rightarrow e = 161 - 127 = 34$ $\quad$ $Significand = 1.11110101101$

90FAD000 = $-1.11110101101 \times 2^{34}$

$X = 1.10110101101 \times 2^{34} + 1.11110101101 \times 2^{34}$ (unsigned addition)

```
c12=1 c11=1 c10=1 c9=1 c8=1 c7=0 c6=1 c5=0 c4=1 c3=1 c2=0 c1=1 c0=0
    1.1 0 1 1 0 1 0 1 1 0 1 +
    1.1 1 1 1 0 1 0 1 1 0 1
  -------------------------
  1 1.1 0 1 0 1 0 1 1 0 1 0
```

$X = 11.1010101101 \times 2^{34} = 1.11010101101 \times 2^{35}$

$e + bias = 35 + 127 = 162 = 10100010$

X = 0101 0001 0110 1010 1101 0000 0000 0000 = 516AD000

✓ X = 01800000 × FAB80000:

01800000: 0000 0001 1000 0000 0000 0000 0000 0000

$e + bias = 00000011 = 3 \rightarrow e = 3 - 127 = -124$ $\quad$ $Significand = 1.0$

01800000 = 1.0×2^{-124}

FAB80000: 1111 1010 1011 1000 0000 0000 0000 0000

$e + bias = 11110101 = 245 \rightarrow e = 245 - 127 = 118$ $\quad$ $Significand = 1.0111$

FAB80000 = -1.0111×2^{118}

$X = (-1.0 \times 2^{-124}) \times (1.0111 \times 2^{118}) = -1.0111 \times 2^{-6}$

$e + bias = -6 + 127 = 121 = 01111001$

X = 1011 1100 1011 1000 0000 0000 0000 0000 = BCB800000

✓ X = 7B390000 ÷ C8C00000:

7B390000: 0111 1011 0011 1001 0000 0000 0000 0000

$e + bias = 11110110 = 246 \rightarrow e = 246 - 127 = 119$ $Significand = 1.0111$

7B390000 = 1.0111001×2^{119}

C8C00000: 1100 1000 1100 0000 0000 0000 0000 0000

$e + bias = 10010001 = 145 \rightarrow e = 145 - 127 = 18$ $Significand = 1.1$

C8C00000 = -1.1×2^{18}

$$X = -\frac{1.0111001 \times 2^{119}}{1.1 \times 2^{18}} = -\frac{1.0111001}{1.1} \times 2^{101}$$

```
           000000001111
11000000 ) 101110010000
             11000000
            101100100
             11000000
             101001000
              11000000
              100010000
               11000000
                1010000
```

Alignment:

$$\frac{1.0111001}{1.1} = \frac{1.0111001}{1.1000000} = \frac{10111001}{11000000}$$

Append x = 4 zeros: $\frac{10111001\mathbf{0000}}{11000000}$

Integer division

$Q = 1111 \rightarrow Qf = 0.1111$

Thus: $X = -0.1111 \times 2^{101} = -1.111 \times 2^{100}$

$e + bias = 100 + 127 = 227 = 11100011$

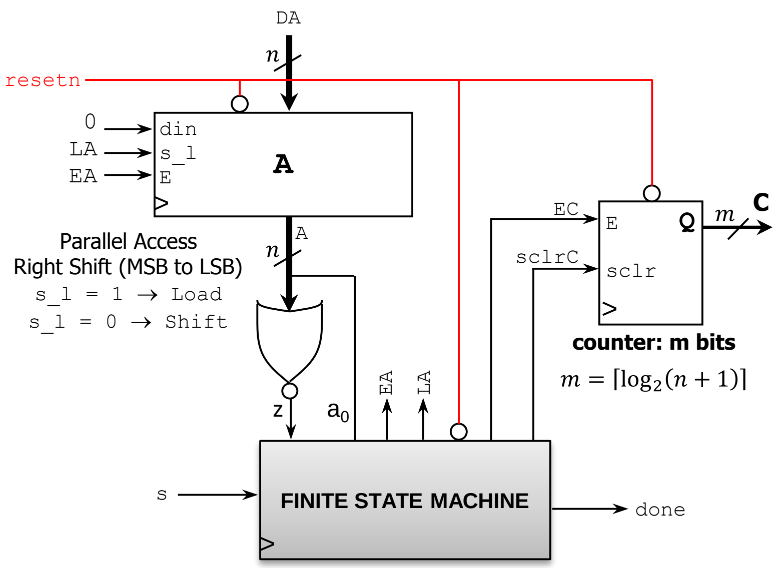
X = 1111 0001 1111 0000 0000 0000 0000 0000 = F1F00000

PROBLEM 4 (32 PTS)

- "Counting 1's" Circuit: It counts the number of bits in register A that has the value of '1'. The digital system is depicted below: FSM + Datapath. Example: For $n = 8$: if $A = 00110110$, then $C = 0100$.
  - ✓ m-bit counter: $sclr$. If $E = sclr = 1$, the count is initialized to zero. If $E = 1, sclr = 0$, the count is increased by 1.
  - ✓ Parallel access shift register: If $E = 1$: $s\_l = 1 \rightarrow$ Load, $s\_l = 0 \rightarrow$ Shift.

- Sketch the Finite State Machine diagram (in ASM form) given the algorithm.
  - ✓ The process begins when s is asserted, at this moment we capture DA on register A. Then the process starts by shifting A one bit at a time. The process is concluded when $A = 0$. The signal $done$ is asserted when we finish counting.
  - ✓ Note: If $A = 0 \rightarrow z = 1$, if $A \neq 0 \rightarrow z = 0$. As A is being shifted, every time $a_0 = 1$, we need to increase the count C.

- Complete the timing diagram (next page) where $n = 8, m = 4$. (12 pts.)



ALGORITHM

```
C ← 0
while A ≠ 0
    if a0 = 1 then
       C ← C + 1
    end if
    right shift A
end while
```

Finite State Machine: